Subcode and sector synchronization for a data processing system

The present invention relates to the data synchronization
between a data processing system and a servo-system for an
5   apparatus for reading from and/or writing to optical recording
media.

In recent years an increasing need has been experienced for
providing consumer electronic devices, like e.g. compact disc
10   players, DVD-players and others, with the capability of
processing audio data stored as audio files in a compressed
(e.g. MP3 or MP3 Pro, WMA (Windows Media Audio)) or
uncompressed (e.g. WAV (Waveform Audio File)) data format on a
storage medium. In contrast to an Audio Processing System
15   according to prior art, which can only read and/or write audio
information according to the Compact Disc Standard (Red Book),
an advanced data processing system (DPS) capable of playing
back and/or recording audio files in a compressed or
uncompressed data format, needs to be able to extract, decode,
20   and convey data related to the file system and the audio files
stored on a storage medium.

A CD (compact disc) decoder IC with embedded compressed audio
decoding shall comprise two primary systems: An advanced data
25   processing system (DPS) and a servo system (performing servo,
data path and control operations). The servo system contains a
master micro controller.

Throughout the description, the following definitions are used:
30
Sector, Subcode address:
All sector addresses and subcode time codes (sections) are
represented in the form min:sec:frame. The minutes field is
represented by "min", the seconds field is represented by
35   "sec", the blocks (sector) or frames (subcode) field is
represented by "frame".

Session:

An area on a disc consisting of a Lead-In Area followed by a Program Area and a Lead-Out Area is called session.

5   Single session disc:

If a disc contains one session, this disc is called a single session disc.

Multisession disc:

10 If a disc contains more than one session, this disc is called a multisession disc.

Sector:

Whenever the term "sector" is used in this document, it has the

15 identical meaning as the term "block" in the CD-ROM system description.

DPS:

(Advanced) Data Processing System. An example for such a data

20 processing system is an advanced audio processing system, a module responsible for the extraction and interpretation of digital data tracks (Digital Tracks, as defined in ECMA-130) in a storage medium, as well as decoding and playback of compressed audio data contained in the digital data tracks.

25

The DPS uses sectors (sector addresses) as its basic data structure while the servo system uses subcode-frames (sections).

30 The ECMA-130 standard provides the following definitions for sector and section of a compact disc:

A sector is the smallest addressable part of the information area that can be accessed independently.

A section (subcode-frame) is defined as a group of 98 F3-frames

35 containing a complete table of control bytes. A table is generated with the control bytes of each subcode-frame

(section). The content of this table is renewed every 1/75 second, which is the same rate at which the subcode-frames are being processed. The table of control bytes of a subcode-frame defines the following channels: p-, q-, r-, s-, t-, u-, v- and w-channel of 96 bits each. The receiving end has particular interest in the q-channel that conveys, among others, information about the absolute time (subcode time code) of the section in the A-FRAC, A-SEC and A-MIN fields. These fields contain the absolute time in 75th of a second as a continuation of the time specified in the A-xxx fields of the q-channel of the preceding section.

The address of a sector is recorded in the sector header. The address of the first sector with user data of an information track is written in the table of contents of the disc. The sector header conveys the address of a sector as an absolute time. At the encoder end, this address is inserted just before the sector enters a Cross Interleaved Reed-Solomon (CIRC) encoder.

According to the ECMA 130 standard, the nominal value of a sector shall be equal to the absolute time recorded in the control bytes of that subcode-frame which is being processed by the 8-to-14 encoder at the time the sync word of the sector enters the CIRC encoder.

It is also stated in the ECMA-130 standard that the address of the sectors has no prescribed relation to the address (subcode time code) of the sections, due to the fact that the mapping of a sector to the sections during recording is implementation dependent (ECMA-130 Standard, clause 16, F1 Frame). For this reason synchronization between sectors and subcode-frames is required at the receiving end.

The decoder IC servo system processes the q-channel of the subcode-frames. The micro controller has access to the q-channel data. Through this it has access to the CD's table of

- 4 -

content. The table of content entries point to the start of
information tracks on the disc, in terms of the absolute time
in the control bytes, with an accuracy of +/- 1 second.
The micro controller has also access to the subcode absolute
5   time code in the q-data field of the information area.
The DPS requests sectors from the micro controller. The micro
controller organizes the task of delivering data to the DPS
with the collaboration of an electronic shock proof micro
controller (ESPMC) block, from which the DPS receives its input
10 data.
The DPS requests sectors using the sector address expressed in
absolute time elapsed since the beginning of the user data
area, in MSF (minutes, seconds, frames) units. The
corresponding information in the servo system is the absolute
15 time code part of the q-channel of a subcode frame. If no
synchronisation between the subcode time code and the sector
address is done, the DPS has to flush the data provided by the
servo system until the requested data (sector) is received,
which is time and memory consuming. If the time difference is
20 known, the servo system can take this information into account
when jumping to a requested time.

US 5,574,704 discloses a header searching method for a data
recording medium, which is capable to reduce the a number of
25 required times to access to a target header, and to shorten
time on being given a time information of the target header and
searching the header. For this purpose a time information of a
sub-code is read out, and then, a time information of a header
is read out. In accordance with a difference between the time
30 information of the sub-code and the time information of the
header, an offset value is calculated and memorized.

It is an object of the present invention to provide an improved
method for synchronizing the subcode time codes and sector
35 addresses of data contained on a recording medium.

According to the invention, the method comprises the steps of:
- sending a number of sectors from the micro controller to the data processing system;
- requesting information about the sector headers of the
5 received sectors from the data processing system;
- calculating the difference between the subcode time codes and the sector addresses using the information about the sector headers; and
- repeating the synchronisation steps for every session on the
10 recording medium.
The method has the advantage that it will speed-up the access of the data processing system to the sector information provided by the servo-system. Since the difference can vary for different sessions on the same disc, this ensures
15 synchronization between the subcode time codes and the sector addresses for all sessions.

According to another aspect of the invention, the method comprises the steps of:
20 - sending a number of sectors from the micro controller to the data processing system;
- requesting information about the sector headers of the received sectors from the data processing system;
- calculating the difference between the subcode time codes and
25 the sector addresses using the information about the sector headers;
- asking the data processing system for a confirmation of sector reception; and
- implementing a continuity counter in the data processing
30 system to check if the expected sectors were received.
By asking the processing system for a confirmation of sector reception and by implementing a continuity counter it is ensured that no sectors were missed, which could otherwise lead to an incorrect synchronization between the subcode time codes
35 and the sector addresses.

Advantageously, the method further comprises the step of storing the sectors in a memory. This allows further access to the sectors by subsequent additional commands from the micro controller.

5

According to the invention, absolute time information conveyed in the sector headers and in absolute time fields of the q-channel of the subcode frame is used for calculating the difference between the subcode time codes and the sector

10   addresses. Using this information the synchronization between the subcode time codes and the sector addresses can easily be achieved.

According to the invention, a communication protocol for the

15   communication between a data processing system and a micro controller defines a set of commands and messages necessary for synchronization between subcode time codes and sector addresses of data contained on a recording medium. Such a communication protocol allows to synchronize between two systems, where one

20   system is working based on subcode information, and the other system is working based on sector information.

Advantageously, the set of commands and messages defined for the communication protocol comprises commands for scanning

25   sectors and for reading sector data, messages for sending information on the sectors and the read sector data, and commands and messages for asking the data processing system for a confirmation of sector reception. From the sector data, especially the sector header, all information necessary for synchronization can be

30   retrieved.

Favourably, a decoder for optical recording media performs a method or uses a communication protocol according to the invention.

35

According to the invention, the inventive method, the inventive
communication protocol, and/or the inventive decoder is used in
an apparatus for reading from and/or writing to optical
recording media.

In the following, the implementation of a communication
protocol according to the invention will be explained in more
detail. Commands are sent from the micro controller to the DPS,
while messages are sent from the DPS to the micro controller.

Synchronization process:
After the servo system has finished reading the table of
content (TOC), in case of single session disc, the micro
controller will start an "analyse mode" on the DPS. In case of
a multisession disc, the analyse mode can be started after the
TOC of each session is read, or after all TOCs are read.

Description of command and Message:
Used command: SCAN <frame><sec><min>
Used message: SCANM
Three different Modes for SCAN, depending on the transmitted
frame:sec:min information are used.

Mode 1:
Requests to get information about the first valid sector which
can be received from the AIN-Mailbox (Audio-In Mailbox).
Before the servo system is sending a SCAN model command to the
DPS, the ESPMC memory will be flushed and then e.g. 4 subcode
frames are stored by the servo system into the ESPMC memory.
The storing is to be understood as follows. In a playback mode
the servo system searches for the specific subcode or jumps to
the corresponding position on the disc. As soon as the subcode
has been found, writing of data to the ESPMC is enabled, the
data coming from an error correction block. When a
predetermined number of subcode-frames, e.g. four subcode-
frames, has been written to the ESPMC, writing is disabled

again. The micro controller does not write data to the ESPMC
directly.
The storage of data is based on the subcode information. By
reading the data in the ESPMC via the DPS the header

5   information (sector) can be accessed and the time difference
between the stored subcode frame and the actually obtained
sector can be calculated.

Function:

10   After receiving the SCAN mode1 command from the servo system,
the DPS starts reading data out of the AIN-Mailbox, while
counting the flushed bytes, until a SYNC word, followed by a
header (1st header, i.e. 12 byte SYNC, followed by 4 bytes of
non flagged data (C2_ERR = 0)) is found. The 1st header

15   (4bytes), followed by 2336 bytes of data will be stored.

The DPS has to check now, if the following SYNC-word and the
following header (2nd header, i.e. 12 byte SYNC, followed by 4
bytes of non flagged data (C2_ERR = 0)) are valid. The SYNC-

20   word and the 2nd header have to be stored to allow subsequent
SCAN mode3 commands to operate on the same ESPMC data.

The command will be aborted by the DPS if the ESPMC memory
becomes empty, or in case a number of byte counter has an

25   overflow.

The DPS will send a SCANM message to the servo system, carrying
information on the 1st and 2nd valid headers, the number of
counted bytes, which were flushed while reading the AIN-Mailbox

30   until the 1st header was found, and an indicator indicating a
number of byte counter overflow.

The number of counted bytes should be normally less than the
number of bytes of one sector (2352). Otherwise, this would be

35   an indication for the servo system that one sync-word,
respectively header was not found.

The information on the 1st and 2nd valid headers comprises 4 byte header information (1st header) as coded on disc and 8 byte subheader information (1st header) as coded on disc, and indicators indicating whether valid 1st and 2nd headers were found.

After receiving the SCANM message, the servo system is able to calculate the time difference (in complete sectors) between subcode time code and sector addresses as follows:

time difference = { [min:sec:frame] subcode time code stored in ESPMC} - {[min:sec:frame] received 1st header information}

Since only complete sectors can be taken into account by the servo system, the DPS has to search for the sync word and has to flush number of bytes when the delta is calculated.
The exact delta will be:

time difference + { (number of byte-12) / 2352}

Mode 2:
Requests to search for sector [min:sec:frame] within the data which can be received from the AIN-Mailbox.
Before the servo system is sending a SCAN mode2 command to the DPS, the ESPMC memory will be flushed and then the data to be analysed is stored into the ESPMC memory.

Function:
After receiving the SCAN mode2 command from the servo system, the DPS starts reading data out of the AIN-Mailbox, while counting the flushed bytes, until a SYNC word, followed by the requested header (1st header) is found. The 1st header (4bytes), followed by 2336 bytes of data will be stored.

The DPS has to check now if the following SYNC-word and the
following header (2nd header) are valid. The SYNC-word and the
2nd header have to be remembered to allow subsequent SCAN mode3
commands to operate on the same ESPMC data.

The command will be aborted by the DPS if the ESPMC memory
becomes empty, or in case a number of byte counter has an
overflow.

The DPS will send a SCANM message to the servo system, carrying
information on the 1st and 2nd valid headers, the number of
counted bytes, which were flushed while reading the AIN-Mailbox
until the 1st header was found, and an indicator indicating a
number of byte counter overflow.

The information on the 1st and 2nd valid headers comprises 4
byte header information (1st header) as coded on disc and 8
byte subheader information (1st header) as coded on disc, and
indicators indicating whether valid 1st and 2nd headers were
found.

After receiving the SCANM message, the servo system is able to
calculate the time difference (number of frames) between
subcode time code and sector addresses as follows:

time difference = {Number of counted bytes mod 2352}

Mode 3:
Requests information about the next sector (2nd valid sector of
the previous scan) which can be received from the AIN-Mailbox.
The servo system sends a SCAN mode3 command to the DPS. This
mode is only applicable, if scan mode1, scan mode2 or scan
mode3 was sent before. If no scan was sent before or if no 2nd
valid header was found before, the error message "command not
available" will be returned.

Function

After receiving the SCAN mode3 command from the servo system, the DPS continues reading data out of the AIN-Mailbox. The data will be stored into. No bytes will be flushed.

5 The header and subheader information of the current sector, which was already read by the previous scan mode1 or scan mode2 command (previous 2nd valid header becomes 1st valid header), will also be stored into.

10 The DPS has to check now if the following SYNC-word and the following header (new 2nd header) are valid. The SYNC-word and the 2nd header have to be stored to allow subsequent SCAN mode3 commands to operate on the same ESPMC data.

15 The command will be aborted by the DPS, if the ESPMC memory becomes empty.

The DPS will send a SCANM message to the servo system, carrying information about the 1st valid header and about whether a 2nd
20 valid header was found.

The information about the 1st valid header comprises 4 byte header information as coded on disc and 8 byte subheader information as coded on disc.

25

Read sector data:

As an option, and also to be flexible for further changes in the CD-Sector formats, the following access to the cd-rom data is proposed. The command and message described below allow to
30 read the data (header and cd-rom data) as stored by previous scan commands.

Description of command and Message:
Used command: RSECDATA (Request Sector Data)
35 Used message: SECDATA

The command, respectively message is only valid if a SCAN
command was sent before. Otherwise an error message "command
not available" will be returned.

5   The RSECDATA command is called with 2 bytes (lower, higher)
indicating the first value which will be requested.

Function
After receiving the RSECDAT command from the servo system, the
10   DPS sends the requested data starting at the requested first
value, to the micro controller using the SECDATA message.

If the first requested value lies outside an allowed range, the
error message "target not available" will be returned.

15

The inventive method is not limited to the communication in a
CD-decoder. The method is applicable to any communication
between two systems, where one system is working based on
subcode information (like the servo system), and the other
20   system is working based on sector information (like the DPS).

For a better understanding of the invention, an exemplary
embodiment is specified in the following description of an
advantageous configuration with reference to the figure. It is
25   understood that the invention is not limited to this exemplary
embodiment and that specified features can also expediently be
combined and/or modified without departing from the scope of
the present invention.

30   Fig. 1   shows an example for the SCAN model communication.

Fig. 1 shows an example for the SCAN model communication. Upon
initialization, the micro controller 1 in a first step 4 has a
number of sectors sent via a memory 2 (e.g. an electronic shock
35   proof micro controller ESPMC) to the DPS 3. In a second step 5
the micro controller 1 asks for a confirmation on sector

reception, which is sent in a further step 6 from the DPS. For
this purpose a continuity counter 7 is implemented in the DPS.
To allow synchronization between the subcode time codes and the
sector addresses, the micro controller in a step 8 requests

5   information about the sector header fields using a SCAN model
command. The DPS sends the requested information using a SCANM
message. The micro controller is now able to calculate 9 the
difference between the subcode time codes and the sector
addresses. In the figure, two further communications steps are

10   shown. Using a first read sector data command RSECDATA1, the
micro controller 1 requests sector data from the DPS 3, which
in response sends the requested data using a first sector data
message SECDATA1. Further data is then exchanged via a second
read sector data command RSECDATA2 and the corresponding sector

15   data response SECDATA2.